## IN THE SPECIFICATION

Please replace paragraph 2 on page 13 lines 6-11 with the following new paragraph:

~~The concepts of sending periods and synchronization are illustrated in Figure 1. The timeline 100 has an origin 102, which denotes the moment when the first wafer is loaded into the cluster tool. The timeline 100 is demarcated in units of one sending period 110. Each demarcation 104 106 108 indicates, respectively, the times at which the second, third, and fourth wafers are loaded into the cluster tool.~~

The concepts of sending periods and synchronization are illustrated in Figure 1. The timeline 100 has an origin 102, which denotes the moment when the first wafer is loaded into the cluster tool. The timeline 100 is demarcated in units of one sending period 110. Each demarcation 104, 106, 108 indicates, respectively, the times at which the second, third, and fourth wafers are loaded into the cluster tool.

Please replace paragraph 3 on page 13 lines 12-23 with the following new paragraph:

~~A principal characteristic of synchronization is periodicity: the present invention ensures that for each task I, I=1, . . ., n, the pick-up times for any wafer undergoing that task are identical. Thus each task i in the cluster tool can be associated with a relative pick-up time denoted $T_i$ where $T_i$ is normalized in units of the sending period. Figure 2 depicts this feature of periodicity. Three wafers, wafer 1 208, wafer 2 210, and wafer 3 212 are depicted on the vertical axis 202. The horizontal line depicts the TIME axis 200. The origin of this axis 201 indicates the time at which wafer 1 is loaded into the cluster tool. The relative pick-up times $T_i$ at task i 200 are identical for each wafer. Because the wafers themselves are introduced at intervals for one sending period, the actual pick-up times are separated by units of one sending period.~~

A principal characteristic of synchronization is periodicity: the present invention ensures that for each task I, I=1, . . ., n, the pick-up times for any wafer undergoing that task are identical. Thus each task i in the cluster tool can be associated with a relative pick-up time denoted $T_i$ where $T_i$ is normalized in units of the sending period. Figure 2 depicts this feature of periodicity. Three wafers, wafer 1, 208, wafer 2, 210, and wafer 3, 212 are depicted on the vertical axis 202. The horizontal line depicts the TIME axis 200. The origin of this axis 201 indicates the time at which wafer 1 is loaded into the cluster tool. The relative pick-up times $T_i$ at task i 200 are identical for each wafer. Because the wafers themselves are introduced at intervals for one sending period, the actual pick-up times are separated by units of one sending period.

Please replace paragraph 1on page 14 lines 1-11 with the following new paragraph:

~~Figure 2 also illustrates a distinction between relative and "actual" or "absolute" pick-up times. The relative pick-up time of a process i is denoted by $T_i$ 204. Since the relative pick-up times is measured from the time a wafer is introduced into the wafer cluster tool, the relative pick-up time is identical for each wafer, wafer 1 208, wafer 2 210, and wafer 3 312. The absolute pick-up time 214 is measured from the moment the first wafer was loaded into the cluster tool 201. Since the wafer are introduced at intervals of one sending period, it follows that for any wafer no. w, the absolute pick-up time of wafer w at module i is~~

~~$(w-1) + T_i$~~

~~This period (w-1) is illustrated in figure 216.~~

Figure 2 also illustrates a distinction between relative and "actual" or "absolute" pick-up times. The relative pick-up time of a process i is denoted by $T_i$ 204. Since the relative pick-up times is measured from the time a wafer is introduced into the wafer cluster tool, the relative pick-up time is identical for each wafer, wafer 1, 208, wafer 2, 210, and wafer 3, 312. The

absolute pick-up time 214 is measured from the moment the first wafer was loaded into the cluster tool 201. Since the wafer are introduced at intervals of one sending period, it follows that for any wafer no. w, the absolute pick-up time of wafer w at module i is

$(w-1) + T_i$

This period (w-1) is illustrated in figure 216.

Please replace paragraph 2 on page 25 lines 11-23 and page 26 lines 1-19 with the following new paragraph:

~~**Step 4 Queuing For Synchronization.** The information in the table 800 of Figure 8 is a summary of load allocation from Step 3. Only the six modules served by main robot C-1, i.e., VPC 802, CT 804, SBC 806, PEBC 808, DEV 810, and HB 812, need to be considered for queuing. The remaining modules should not experience conflicts since each has dedicated robot, i.e., an IBTA, serving them. In the Target Column 814, the target timing profile is set for the 6 modules. For each of the modules listed in the table, a corresponding value is set for $\tau^*$, where $\tau^*$ is an updated value for $\tau$ which eliminates conflicts between tasks for robots. Since only 3 of these 6 modules, SBC 806, DEV 810, and HB 812 are in conflict, only two, DEV and HB, need to have timing targets different from the original values for $\tau$ prescribed by the recipe. The timing targets are listed in column 814, and are set such that the timing interval between any pair of the 6 modules is larger than robot transport time (=6/45 ~ 0.1333). The differences between the target and the originally prescribed timing profiles are referred to as the gaps and are computed the column 816. These are shown pictorially in the graph 818 adjacent to the table. Another objective of the "synchronous alogorithm" is to ensure that no delays are introduced at critical process steps. In this example, critical process steps are step 3 804, step 4 806, and step 7 809. No queues should be added to modules corresponding to these steps, i.e., the target timing for~~

~~these modules should be the same as prescribed values. The gaps computed in the Gap column 816 may now be substituted in to Equation (3) to solve for the queues that will close these gaps. However, to ensure zero delays at the critical process modules, the matrix 818 relating gaps and queues must be modified per Equation (4) to generate a modified matrix 820. Pre-multiplying the gaps from the Gap column 816 with the inverse of the modified matrix 822 produces the queues needed to be close the gap 824. The solution for the queues is transferred to a Que Column 826. The solution, which is in units of the sending period, is converted to actual time in an Actual Que Column 828.~~

**Step 4 Queuing For Synchronization.** The information in the table 800 of Figure 8 is a summary of load allocation from Step 3. Only the six modules served by main robot C-1, i.e., VPC 802, CT 804, SBC 806, PEBC 808, DEV 810, and HB 812, need to be considered for queuing. The remaining modules should not experience conflicts since each has dedicated robot, i.e., an IBTA, serving them. In the Target Column 814, the target timing profile is set for the 6 modules. For each of the modules listed in the table, a corresponding value is set for $\tau^*$, where $\tau^*$ is an updated value for $\tau$ which eliminates conflicts between tasks for robots. Since only 3 of these 6 modules, SBC 806, DEV 810, and HB 812 are in conflict, only two, DEV and HB, need to have timing targets different from the original values for $\tau$ prescribed by the recipe. The timing targets are listed in column 814, and are set such that the timing interval between any pair of the 6 modules is larger than robot transport time (=6/45 ~0.1333). The differences between the target and the originally prescribed timing profiles are referred to as the gaps and are computed the column 816. These are shown pictorially in the graph 818 adjacent to the table. Another objective of the "synchronous alogorithm" is to ensure that no delays are introduced at critical process steps. In this example, critical process steps are step 3, 804, step 4, 806, and step 7, 809. No queues should be added to modules corresponding to these steps, i.e., the target timing for

these modules should be the same as prescribed values. The gaps computed in the Gap column 816 may now be substituted in to Equation (3) to solve for the queues that will close these gaps. However, to ensure zero delays at the critical process modules, the matrix 818 relating gaps and queues must be modified per Equation (4) to generate a modified matrix 820. Pre-multiplying the gaps from the Gap column 816 with the inverse of the modified matrix 822 produces the queues needed to be close the gap 824. The solution for the queues is transferred to a Que Column 826. The solution, which is in units of the sending period, is converted to actual time in an Actual Que Column 828.

Please replace paragraph 2 on page 33 lines 11-22 and page 34 lines 1-3 with the following new paragraph:

~~By the *n*th wafer, the cluster tool will be fully populated with wafers. For every wafer exiting the cluster tool, there is a wafer entering to replenish it. All process and transport tasks performed on the wafers occur in a periodic fashion, the periodicity being delimited by the send period. When the system reaches this stage, it is said to be in steady state. The numbers of wafers *n* required to be fully populate the cluster tool and ramp up to a steady periodic state is given by the expression~~

$$n = 1 + INT\left(\frac{\pi}{SP}\right); \qquad \text{(A)}$$

~~where $\pi$ is the total process and transport time performed on a wafer as indicated in Figure 9-902. The symbol INT(•) denotes a function that rounds a number down to the nearest integer. The above equation also applies for the number of wafers for a lot to ramp down from a steady state and fully empty out the cluster tool. In the steady state, every wafer has identical process and~~

~~transport tasks performed on it at identical time intervals. There is therefore no need to keep track of the movement of wafers inside the cluster tool.~~

By the *n*th wafer, the cluster tool will be fully populated with wafers. For every wafer exiting the cluster tool, there is a wafer entering to replenish it. All process and transport tasks performed on the wafers occur in a periodic fashion, the periodicity being delimited by the send period. When the system reaches this stage, it is said to be in steady state. The numbers of wafers *n* required to be fully populate the cluster tool and ramp up to a steady periodic state is given by the expression

$$n = 1 + INT\left(\frac{\pi}{SP}\right); \qquad \text{(A)}$$

where $\pi$ is the total process and transport time performed on a wafer as indicated in Figure 9, 902. The symbol INT(•) denotes a function that rounds a number down to the nearest integer. The above equation also applies for the number of wafers for a lot to ramp down from a steady state and fully empty out the cluster tool. In the steady state, every wafer has identical process and transport tasks performed on it at identical time intervals. There is therefore no need to keep track of the movement of wafers inside the cluster tool.